
Subject: [RFC] [PATCH 1/4] uid_ns: introduce inode uid check helper

Posted by [serue](#) on Tue, 07 Nov 2006 04:19:15 GMT

[View Forum Message](#) <> [Reply to Message](#)

Uid namespaces will require that when a tasks' permission to an inode is checked, not just the uid, but also the namespace is checked. Since this is a pervasive change, let's start by introducing a helper without making any semantic changes, so we can make the semantic change in one place.

Signed-off-by: Serge E. Hallyn <serue@us.ibm.com>

fs/namei.c | 11 ++++++-----
include/linux/fs.h | 5 +++++
2 files changed, 11 insertions(+), 5 deletions(-)

f3afe1adf8ebb6d2e7975dada086df5e9ea36d3c

diff --git a/fs/namei.c b/fs/namei.c

index ede2761..522ec89 100644

--- a/fs/namei.c

+++ b/fs/namei.c

@@ -184,7 +184,7 @@ int generic_permission(struct inode *ino
{
 umode_t mode = inode->i_mode;

- if (current->fsuid == inode->i_uid)

+ if (inode_task_same_uid(inode, current))

mode >= 6;

else {

if (IS_POSIXACL(inode) && (mode & S_IRWXG) && check_acl) {

@@ -436,7 +436,7 @@ static int exec_permission_lite(struct i

if (inode->i_op && inode->i_op->permission)

return -EAGAIN;

- if (current->fsuid == inode->i_uid)

+ if (inode_task_same_uid(inode, current))

mode >= 6;

else if (in_group_p(inode->i_gid))

mode >= 3;

@@ -1360,9 +1360,9 @@ static inline int check_sticky(struct in

{

if (!(dir->i_mode & S_ISVTX))

return 0;

- if (inode->i_uid == current->fsuid)

+ if (inode_task_same_uid(inode, current))

return 0;

```

- if (dir->i_uid == current->fsuid)
+ if (inode_task_same_uid(dir, current))
    return 0;
    return !capable(CAP_FOWNER);
}
@@ -1572,7 +1572,8 @@ int may_open(struct nameidata *nd, int a

/* O_NOATIME can only be set by the owner or superuser */
if (flag & O_NOATIME)
- if (current->fsuid != inode->i_uid && !capable(CAP_FOWNER))
+ if (!inode_task_same_uid(inode, current) &&
+     !capable(CAP_FOWNER))
    return -EPERM;

/*
diff --git a/include/linux/fs.h b/include/linux/fs.h
index 4090d9d..699c7b5 100644
--- a/include/linux/fs.h
+++ b/include/linux/fs.h
@@ -2157,5 +2157,10 @@ static inline void free_secdata(void *se
{ }
#endif /* CONFIG_SECURITY */

+static inline int inode_task_same_uid(struct inode *ino,
+ struct task_struct *tsk)
+{
+ return (ino->i_uid == tsk->fsuid);
+}
#endif /* __KERNEL__ */
#endif /* _LINUX_FS_H */
--
1.1.6

```

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>
